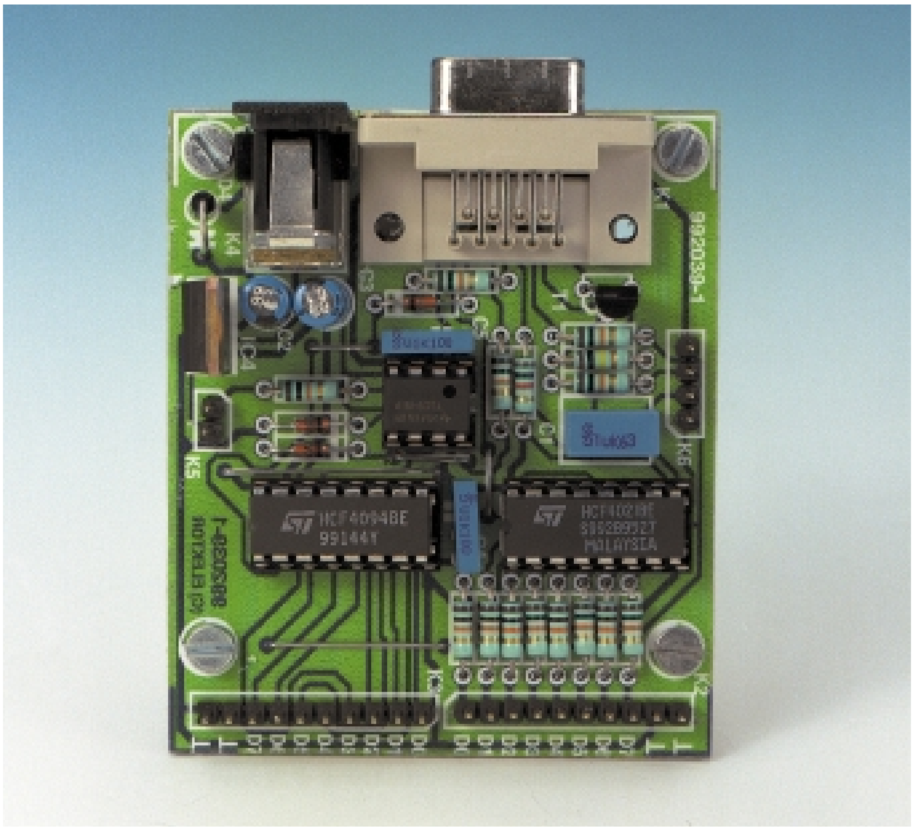
An interface between the PC and the external world does not have to be complicated. This general-purpose Windows interface needs only three ICs, and it works directly from the serial interface. In addition to eight digital outputs and eight digital inputs, it has an analogue input and a frequency-measurement input. The project is complete with easy-to-use Windows software.

Design by B. Kainka

universal interface for Windows

making measurements with Uniface & Compact



The universal interface works directly from the PC's serial interface. It does not employ the asynchronous RS232 standard, but is instead operated under direct control of the serial interface lines. Since the COM interface of the PC has only three output and five input lines, shift registers are used to transfer data serially. A data line and a clock line are necessary for each data transfer channel. Data transfers are not restricted to any given rate, so fast and slow computers can be equally well employed.

The software for the interface was originally written in Turbo Pascal, and could not be directly run under Windows. However, since starting out with the project, the author developed a solution for the Windows platform. The software and the interface described here a Windows 95/98/NT DLL that in principle supports the serial interface, the parallel or printer port, the joystick interface, the sound card and the video card. Here, we concentrate on serial communications.

Input and output shift registers

Figure 1 shows the schematic diagram of the interface. A 4094 CMOS shift register provides eight output lines. The PC can output eight data bits via the RS232 lines TXD (data) and DTR (clock), and then connect these bits to the outputs by applying a pulse to the RTS (strobe) line. The outputs can deliver up to around 5 mA and can directly drive LEDs via series resistors.

The type 4021 shift register makes eight input lines available for reading by the PC. A strobe pulse on the RTS line captures the states of the inputs, which are then read via the DCD line by a series of eight clock pulses applied to the DTR line. The inputs are protected against voltage surges by 10 kΩ series resistors. They have a high input impedance and exhibit random levels when left disconnected.

The simple and inexpensive A/D converter IC, type TCL549, is also addressed as a shift register. It provides its data via the DSR line. It provides an analogue input with a measurement range from 0 to +5 V, with eight-bit resolution. The analogue input is also protected against overvoltage by a 10 kΩ resistor.

The frequency-measurement input works via a small buffer amplifier connected to the CTS input line. The PC can directly read and count pulses. There is a d.c.-coupled input for static signals, pulse-length measurements and low-frequency measurements. There is also an a.c.-coupled input for frequencies up to around 50 kHz. The highest measurable frequency is strongly dependent on the PC used.

COMPONENTS LIST

Resistors:

R1-R5,R8-R15 = 10kΩ

R6 = 100kΩ

R7 = 1kΩ

Capacitors:

C1 = 1µF MKT (Siemens)

C2,C3 = 10µF 63V radial

C4,C5 = 100nF ceramic

Semiconductors:

D1,D2,D3 = zener diode 4V7, 500mW

D4 = 1N4001

T1 = BC547B

IC1 = TLC549CP or -IP (Texas Instruments)

IC2 = 4021

IC3 = 4094

IC4 = 7805 or LP2950

Miscellaneous:

K1 = 9-way sub-D socket (female), angled pins, PCB mount

K2,K3 = 10-way SIL pinheader

K4 = mains adaptor socket, PCB mount

K5 = ⌉2-way pinheader

K6 = 4-way (2x2) pinheader

PCB, order code **992039-1** (see Readers Services page)

Disk, 'Compact' control software, order code **996034-1** (see Readers Services page)

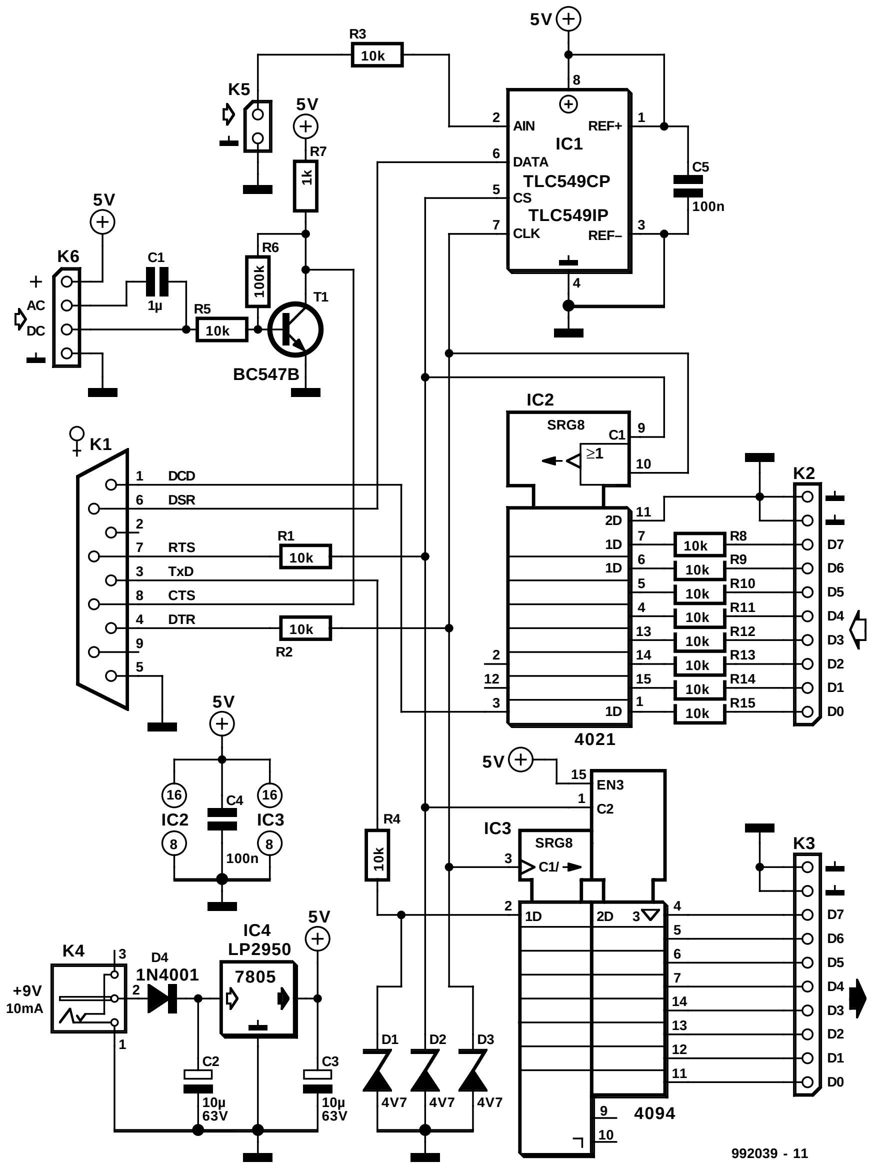


Figure 1. The Windows interface.

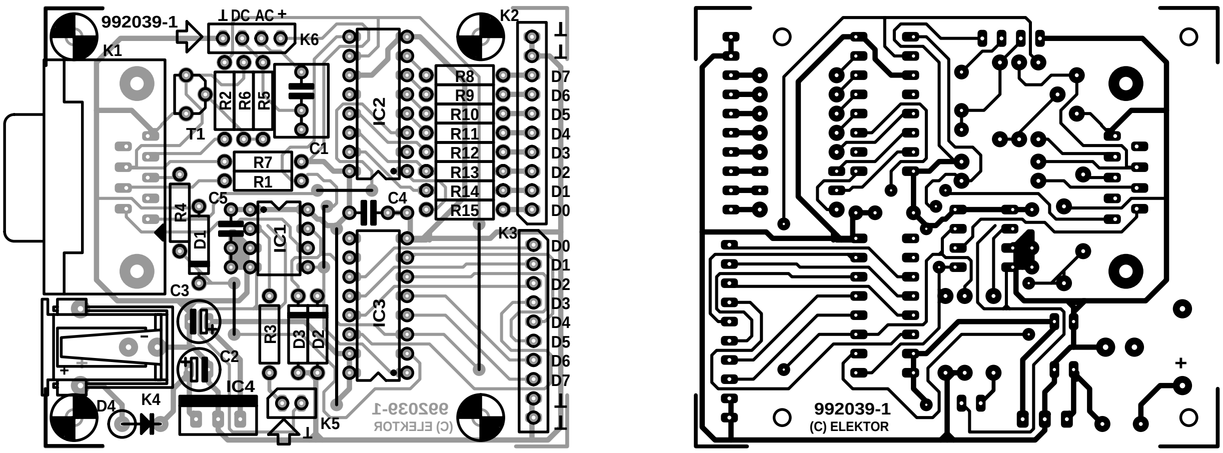
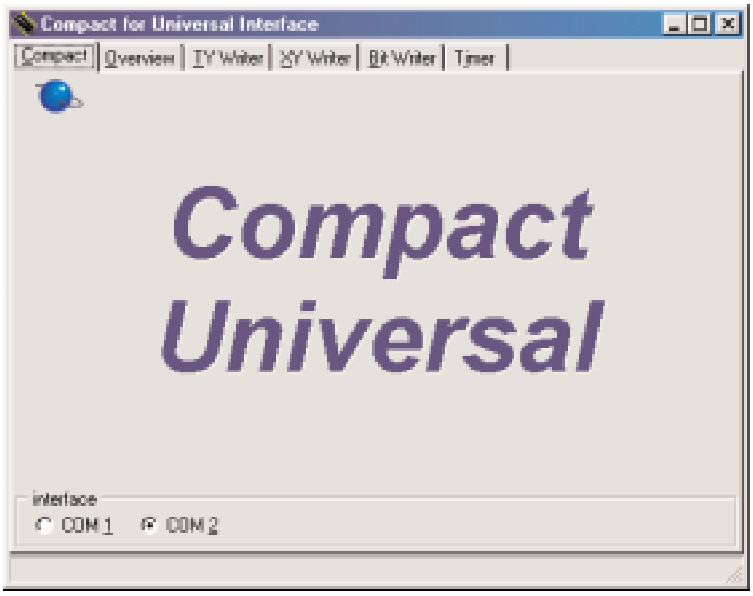
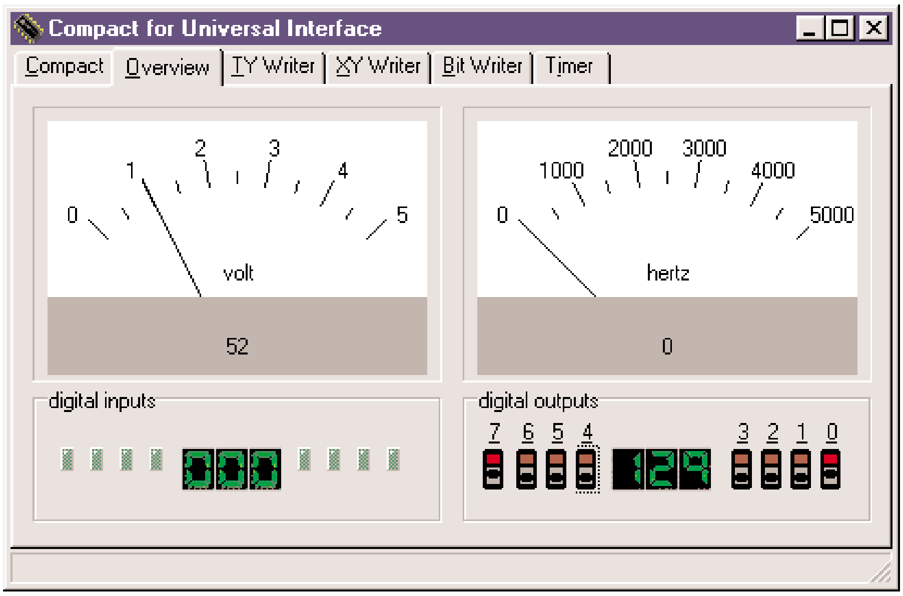


Figure 2. The printed circuit board.



interface
○ COM 1 ● COM 2

Figure 3. The Compact Universal program.



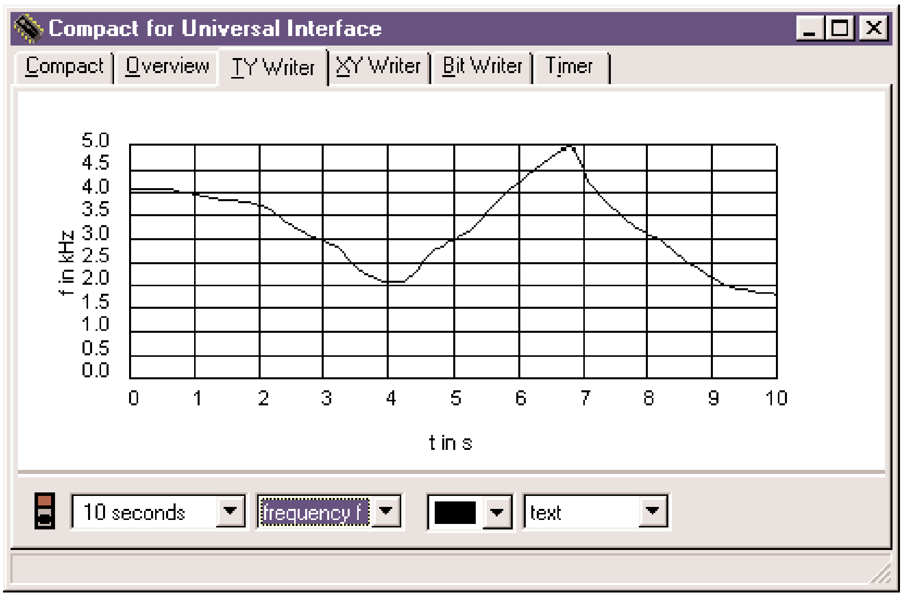
volt: 52

hertz: 0

digital inputs: 000

digital outputs: 129

Figure 4. All inputs and outputs displayed directly.



10 seconds | frequency f | text

Figure 5. The measured value plotted versus time.

With a 200-MHz Pentium, 5 kHz can be reliably achieved under Windows 95. The program 'Compact' can display frequencies up to 5 kHz. With faster PCs, or if the measurements are made under DOS, significantly higher frequencies are possible.

Power is provided by a simple mains adapter having an output voltage of 9 to 12 V, combined with a 5 V regulator IC on the board. A diode connected to the power input jack protects against reverse-polarity connections. **Figure 2** shows the copper track layout and component mounting plan of the printed circuit board. The construction of the board should not present any difficulties.

Compact: a universal measurement program

A versatile application program, called Compact, has been developed for general use in the electronics lab. It is available in English, German, French and even Dutch-language versions. It drives all inputs and outputs of the interface and allows measurements to be taken in a flexible manner. The user can choose the COM port upon starting the program. The simplest and least inexpensive way to obtain Compact is undoubtedly by downloading it from Elektor's Internet home page (*www.elektor-electronics.co.uk*). This will be possible as of 1 December 1999.

After the program is started, a menu with several options appears (see **Figure 4**). The **Overview** function provides the user with direct access to all inputs and outputs. The two analogue inputs, for voltage and frequency, are displayed as pointer instruments. Their values are also displayed digitally, which in the case of the voltage measurement means the value of the data byte provided by the A/D converter. Both instruments can be used directly for experiments and troubleshooting.

In addition to the analogue inputs, the operator has access to the digital inputs and outputs of the universal interface. The eight digital inputs are displayed in the form of virtual LEDs, and also digitally as a byte whose value ranges from 0 to 255. This allows the state of a digital circuit to be directly monitored. The levels of the digital output lines can be altered using virtual slide switches. A digital display shows the state of the combined output bits as a byte value.

The **TY Writer** function plots the magnitude of one or both of the analogue inputs versus time (see **Figure 5**). This simplifies the observation of time-varying values. The measurement interval can be selected within wide limits, ranging from 1 second to 24 hours. Possible applications include long-term measurements, such as recording the charge and discharge characteristics of battery packs. Weather observations using simple sensors are another possibility.

Measured analogue quantities are frequently mutually dependent. In such cases, the **XY Writer** function can be useful (see **Figure 6**), with which the voltage can be plotted against the frequency. This type of presentation is well suited for displaying frequency-response characteristics. The signal from a function generator is used to drive both the frequency-measurement input of the interface and the input of the circuit under test. The output signal from the circuit under test is passed through a measurement rectifier to the voltage input of the interface. The frequency range of interest can be manually scanned in order to generate the response characteristic. **Figure 7** shows the resonant circuit whose frequency response is displayed in **Figure 6**. It is possible to measure the resonant frequency and the Q factor of the inductor. This procedure is also suitable for indirectly measuring the value of the inductance, and for comparing the Q factors of coils having the same inductance.

The **Bit Writer** function implements a **logic analyzer** that is suitable for purely digital applications (see **Figure 8**). The states of up to eight digital signals can be directly observed over a selectable time interval. This function is helpful in troubleshooting digital circuits, for example. The eight digital inputs can be directly connected to up to eight test points in the circuit.

An additional useful function of the program is the **Timer** function. This provides precise time measurements of a pulse waveform connected to the CTS input. As shown in **Figure 9**, a switch connected to earth, with a pull-up resistor to +5 V, can also be connected to this input to make a manually-operated stop watch. The same circuit can be used to measure frequencies using a reed switch.

Programming

The Compact program is naturally only one of the possible applications of the universal interface. Many readers will surely find it interesting to drive the interface using their own programs. **Listing 1** shows in principle how this can be done. The procedure Swap is written

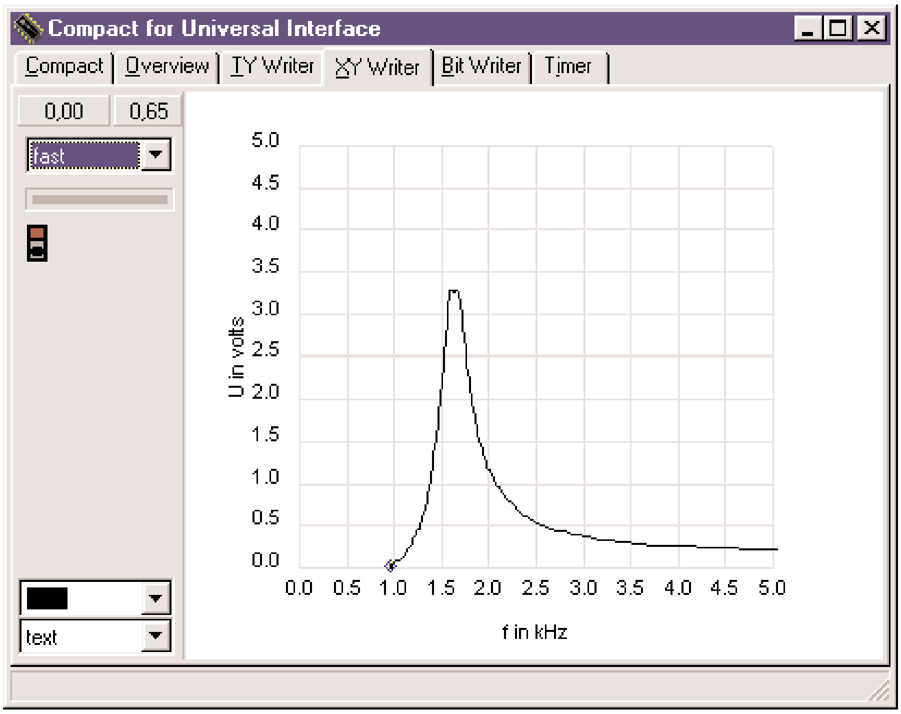


Figure 6. The XY Writer function, and the frequency-response characteristic of a resonant circuit.

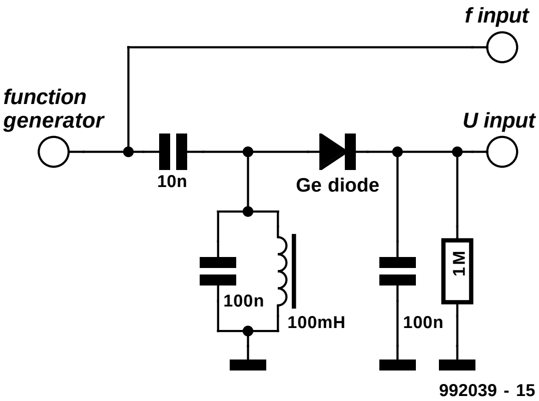


Figure 7. Many parameters of the resonant circuit can be determined using Compact.

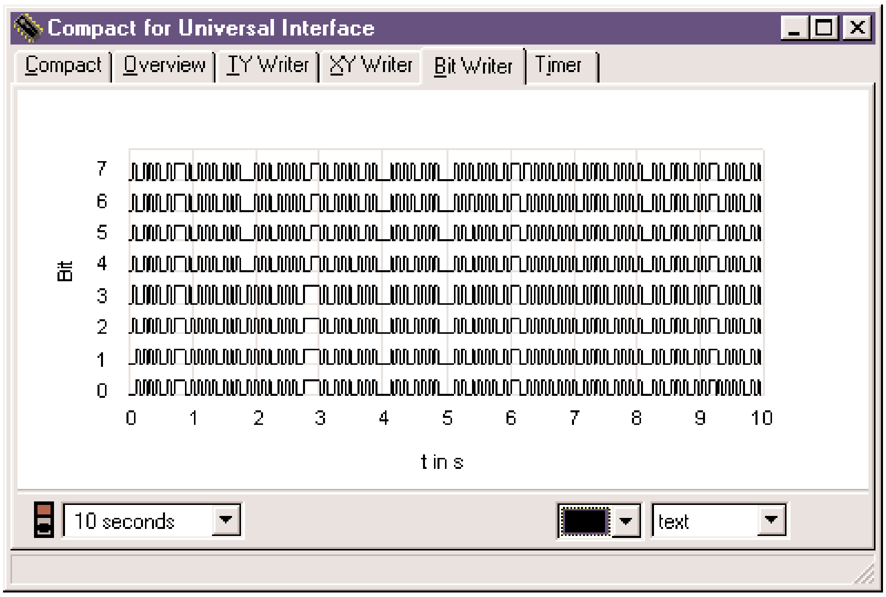


Figure 8. The Bit Writer function can be used as a logic analyzer.

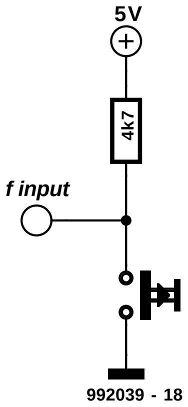


Figure 9. An input switch connected to earth.

in Delphi and services both shift registers and the A/D converter on the circuit board. Data are sent to the output register and read from the input register using the same clock signal. The procedure employs the three global variables Dout (output data for the 4094), Din (input data from the 4021) and Ain (the measurement value from the TLC549).

This program sample also illustrates the application of some important DLL functions to the serial interface. The code for PORTS.DLL has been specially written by H. J. Berndt. The procedures TSD, DTR and RTS can be used for data outputs, and the procedures CTS, DSR, RI and DCD for data input. When the program is started, the interface must be opened using the command

`OpenCom(pchar(‘com2:9600,N,8,1’)).`

The interface parameters, such as the baud rate, are arbitrary. Nonetheless, `OpenCom` registers the usage of the interface with the Windows system. Without this, communication with the interface would not be possible.

The universal interface can of course also be operated under DOS. **Listing 2** shows a suitable Turbo Pascal procedure. Access to the interface lines occurs here via direct port commands.

```
procedure Swap;
var Stelle, StelleAD, n, m  : Integer;
begin
  RTS (1);               { Strobe on }
  Delayus(20);           { Delay }
  RTS (0);               { Strobe off }
  Stelle := 1;
  StelleAD := 128;
  Din := 0;
  Ain := 0;
  for n:=1 to 8 do begin
    if ((Dout AND Stelle) > 0) then
      TXD (1)            { output data }
      else TXD (0);
    if DCD = 1           { read data }
       then Din := Din + Stelle;
    if DSR = 1 then      { read A/D }
       Ain := Ain + StelleAD;
    DTR (1);             { Clock on }
  Delayus(20);           { Delay }
    DTR (0);             { Clock off }
    Stelle := Stelle * 2;
    StelleAD := StelleAD div 2;
  end;
  RTS (1);               { Strobe on }
  Delayus(20);           { Delay }
  RTS (0);               { Strobe off }
end;
```

Listing 1. Sample Delphi driver program.

```
procedure Swap;
var Stelle, StelleAD, n, m  : Integer;
begin
  Port [BA+4] := 2;                        { Strobe on (RTS) }
  for m:=1 to 10 do;                       { Delay }
  Port [BA+4] := 0;                        { Strobe off (RTS) }
  Stelle := 1;
  StelleAD := 128;
  Din := 0;
  Ain := 0;
  for n:=1 to 8 do begin
    if ((Dout AND Stelle) > 0) then
      Port [BA+3] := 64                    { Output data (TXD) }
      else Port [BA+3] := 0;
    if (Port[BA+6] AND 128) = 128          { Read data (DCD) }
       then Din := Din + Stelle;
    if (Port[BA+6] AND 32) = 32 then       { Read A/D (DSR) }
       Ain := Ain + StelleAD;
    Port [Ba+4] := 1;                      { Clock on (DTR) }
    Stelle := Stelle * 2;
    StelleAD := StelleAD div 2;
    Port [BA+4] := 0;                      { Clock off (DTR) }
  end;
  Port [BA+4] := 2;                        { Strobe on (RTS) }
  for m:=1 to 10 do;
  Port [BA+4] := 0;                        { Strobe off (RTS) }
end;
```

Listing 2. Sample DOS driver, using Turbo Pascal 6.

Additional applications

The interface can be used not only as a general-purpose laboratory instrument, but also for quite varied applications, including such things as running-light displays, experimental traffic-signal controllers, alarm installations and small weather stations.

The universal interface may also be used in a test system for automatic verification of digital ICs in the 4000 series. A special program generates bit patterns that are applied to the inputs of the IC under test, and also verifies the output states. Tables that describe the correct behaviour could be added automatically for other types of ICs. In addition to the purely digital functions, the tester would then verify the current consumption to test for insulation faults.

(992039-1)